修 士 論 文 概 要 書

CD

2012 年 2 月提出　　学籍番号 5110B121–9

専門分野	情報理工学専攻	氏 名	村松 聖也	指 導 教 員	戸川 望 印
研究指導	情報システム設計				
研 究 題 目	携帯端末を対象とした 低負荷マルウェア検知システムとその評価に関する研究				

1 序論

近年スマートフォンの増加により，個人情報やビジネスデータを保存する機会や，金銭の授受を伴うサービスが増加した．利便性が向上する一方，マルウェアによるデータや金銭の不正取得等の悪用もされ始めた．実際 2004 年に，モバイルマルウェアが初めて発見された．以降，Symbian 端末や android 端末を対象としたモバイルマルウェアを中心に，2011 年には千種類以上発見されている．マルウェア増加とともに，悪意も強くなっており， 大きな脅威になると予想される．

以上の背景より，携帯端末へのセキュリティシステムの導入は将来不可欠であると考え，侵入検知システムに着目する．先行研究も含め携帯端末への IDS の導入は，携帯端末の処理能力の低さにより，端末への負荷や検知精度が問題となる．本研究では，低負荷かつ高い検知精度を保つ，携帯端末を対象としたマルウェア検知システムの提案を行う．そして，評価実験や考察により，提案システムの有用性を示す．

2 侵入検知システム (IDS)

IDS とは，コンピュータやネットワーク上への侵入・攻撃の特徴的なパターンや兆候を検知し，ユーザや管理者に通知する機能を持つシステムである．

監視対象の違いでホスト型IDS とネットワーク型IDS がある．ホスト型 IDS は，監視するコンピュータにインストールされ，OS やアプリケーションの動作やログを観察する IDS である．ネットワーク型 IDS は，特定のネットワークセグメントに流れるパケットを監視する IDS である．また，検知方法の違いでシグネチャ型検知とアノマリ型検知がある．シグネチャ型検知は，シグネチャと呼ばれる侵入・攻撃の特徴を収録したファイルを用いて，監視対象とパターンマッチを行い，一致したものを侵入として検知する手法である．アノマリ型検知は，機械学習等で生成した正常パターンから一定以上外れたものを異常として検知する手法である．

3 モバイルマルウェア検知の先行研究

研究当初，バッテリ消費量を主な特徴とするアノマリ型検知が提案された．事前に取得したバッテリ消費量の正常パターンを基に，異常を検知する手法である．同時に，端末・サーバ協調型の手法が提案された．端末上で取得した監視データをサーバ上で検査する手法であり，シグネチャ型・アノマリ型問わず適用される．

近年，研究レベルでは，文献 [2] を代表とする，システムコール等の特徴データの抽出と，機械学習により正常パターンを生成するアノマリ型検知が主流である．

バッテリに着目する手法では検知精度が，端末・サーバ協調型の手法ではコストや導入容易性が，文献 [2] を代表とする手法では負荷と誤検知が問題点である．

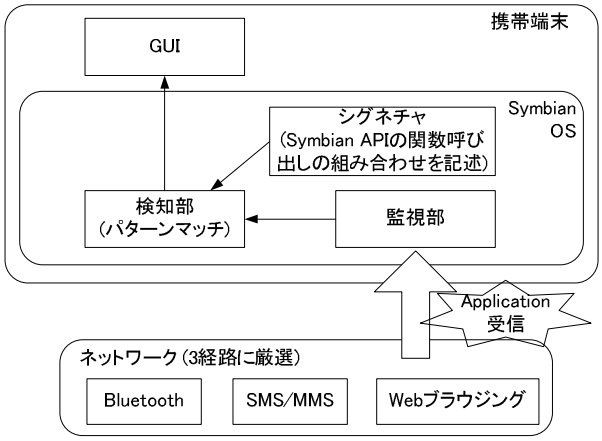

図 1: 提案システムの構成.

4 携帯端末を対象とした低負荷マルウェア検知システム

提案システムのベースは，シグネチャ検知を行うホスト型のソフトウェアである．提案システムの設計では，一例として Symbian OS を用いる．低負荷かつ高い検知精度を保つシステム実現のため，監視対象の厳選と，検知アルゴリズムの考案を行う．

4.1 監視対象の厳選

携帯端末の全通信手段からのマルウェア侵入を監視するのではなく，Bluetooth 通信，SMS/MMS[1]，Web ブラウジングの 3 点からの侵入を監視する．

監視対象の厳選により，上記以外の経路から侵入するマルウェアを検知対象としないため，シグネチャの数を削減し，パターンマッチ処理の負荷を低減できる．また，提案システムが検知処理を行う回数を減らせる．

4.2 検知アルゴリズム

Symbian C++の API で定義された関数呼び出し[2]に着目する．外部からのアプリケーション受信時，受信したアプリケーションのソースコードを取得する．その後，予め定義したシグネチャと，取得したソースコードのパターンマッチを行う．シグネチャには，マルウェアのタイプ毎に Symbian C++の API で定義された関数呼び出しの組み合わせが記述されている．マルウェアであると判断された場合は，GUI に警告を表示する．ここで，提案システムの構成を図 1 に示す．

本検知アルゴリズムの利点は，ソースコードに着目し，OS 固有の API の関数呼び出しに着目することで，バイナリファイルを用いたパターンマッチに比べ，シグネチャに必要な情報量を削減し，パターンマッチ処理の負荷を低減できる．また，ソフトウェア侵入時の

[1] SMS:ShortMessageService,MMS:MultimediaMessageService.
[2] Symbian C++の API の関数呼び出しは，数千種類存在.

みシステムが動作すればよく，機械学習等が必要ないため，システムの動作回数が減ることも利点である．

4.3 提案手法の裏付け

モバイルマルウェアの感染経路には，SMS/MMS, Bluetooth，Web ブラウジング，Email，赤外線通信，USB/PC 接続，MemoryCard/SIMCard 等がある．文献 [1] によると，約 7 割が SMS/MMS，Bluetooth, Web ブラウジングを介する．MemoryCard/SIMCard を介するマルウェアも多いが，利用回数は僅かであるため，監視対象から除外できると考える．

Symbian OS では，アプリケーションのインストール時に，ソースコードを含む SIS ファイルを用いるため，提案検知アルゴリズムによる検知が可能である．また，通常のシグネチャ型はマルウェアのバイナリを基にパターンマッチを行う．しかし，提案検知アルゴリズムでは，複数のバイナリ列に相当する意味を持つ Symbian C++の API の関数呼び出しに着目することで，1 つのシグネチャあたりの情報量の削減が期待できる．

5 提案マルウェア検知システムの実装

提案システムは C++で PC 上に実装する．検知対象ファイルは，SIS ファイルからソースコードが展開されるディレクトリである，`\SYSTEM\APPS\`アプリ名`\src` に置き，検知を開始する．また，シグネチャは signature.txt というファイルに記述する．システムの処理フローは，ソースコードの取得，ソースコードを 1 つにまとめる，シグネチャファイルの読み込み，パターンマッチ，GUI への表示という流れである．

6 提案マルウェア検知システムの評価

提案システムの検知精度とパターンマッチ処理の負荷低減を評価実験で示す．また，システムの動作回数削減による負荷低減を考察で示す．評価実験環境は表 1 に示す．評価実験では，Bluetooth を介して感染する Symbian マルウェアである，Cabir を用いる．

6.1 評価実験

評価実験の前提条件を述べる．Symbian マルウェアが約 400 種類であることと，監視対象の厳選により約 7 割のマルウェア検知に厳選できることより，監視対象の厳選後では元の約 400 種類のシグネチャを約 280 種類まで削減できる．また，文献 [2] によると，マルウェア検知の特徴量を 20 種類〜40 種類と設定している場合が多い．本手法では，関数呼び出しを特徴量とみなす．通常のシグネチャ検知ではバイナリを用いるが，提案検知アルゴリズムでは，複数のバイナリ列に相当する意味を持つ Symbian の API の関数呼び出しに着目するため，特徴量も 50% 以下に削減できると考える．通常のシグネチャ検知での特徴量を 40 種類と仮定することで，提案検知アルゴリズム適用後の 1 つのシグネチャにおける関数呼び出し数を 20 種類とする．

実験 1：検知精度に関する実験

提案システムを用いて，Cabir と正常なサンプルプログラムのパターンマッチを行った．結果，Cabir をマルウェア，サンプルプログラムを正常であると判定した．

実験 2：パターンマッチの負荷に関する実験

提案システム (シグネチャ280 種類，特徴量 20 種類)，監視対象の厳選によるシグネチャ削減のみ (シグネチャ280 種類，特徴量 40 種類)，検知アルゴリズムによるシグネチャ削減のみ (シグネチャ400 種類，特徴量 20 種類)，シグネチャ削減なし (シグネチャ400 種類，特徴量 40 種類) の各システムで，Cabir 検知時のパターンマッチ処理の負荷を測定した．結果は表 2 に示す．提案手法とシグネチャ削減なしを比較すると，提案手法では，実行時間が 17.3% ，最大メモリ消費量が 7.6% ，CPU 使用率が 4.68% 削減された．

表 1: 評価実験環境.

OS	Windows 7 Professional 32bit
CPU	Intel Core 2 Duo 1.4GHz
メモリ (RAM)	2GB
コンパイラ	GNU G++ compiler

表 2: 各システムでのパターンマッチ処理の負荷.

	実行時間 (sec)	最大メモリ消費量 (MB)	CPU 使用率 (%)
提案手法	0.186	4.016	29.54
監視対象厳選による削減のみ	0.203	4.203	31.37
検知アルゴリズムによる削減のみ	0.212	4.109	32.39
signature 削減なし	0.225	4.347	34.22

6.2 考察

監視対象の厳選によりシステムの動作回数が削減される．また，アプリケーション受信時のみの動作に限定することで，機械学習等を用いるアノマリ型に比べ，システムの動作回数が削減される．このことと実験結果から，提案システムの動作回数は少なく，動作時の負荷も低いと言える．また，既存マルウェアを正確に検知し，誤検知も少ないと言える．導入も容易であり，実用的なマルウェア検知システムであると考える．一方，課題点としては，未知マルウェアの検知や，侵入を許してしまったマルウェアへの対策が挙げられる．

本研究では，Symbian OS を対象としたが，Android 等の他の OS への拡張も同様の手法で可能である．

7 結論

本研究では，携帯端末を対象とした低負荷マルウェア検知システムを提案し，評価により有用性を示した．

今後の課題としては，PC 上ではなく実機用に提案システムを実装し，評価データを取得することが挙げられる．また，実際の実用化のためには，シグネチャを完成させる必要がある．

参考文献

[1] A. D. Schmidt and S. Albayrak, “Malicious software for smartphones,” DAI Laboratory, Berlin, Tech. Rep. TUB-DAI 02/08-01, Feb. 2008.

[2] A. Shabtai and Y. Elovici, “Applying behavioral detection on android-based devices,” *Mobile Wireless Middleware, Operating Systems, and Applications*, vol. 48, part 5, pp. 235–249, 2010.